VERIFICATION OF TRANSLATION

I, the below named translator, hereby declare:

that my name and my post office address are as stated below; and

that I am knowledgeable in the English and Korean languages and that I believe the following is a true and complete translation into the English language of Korean Patent Application No. 10-2000-0087576 filed with the Korean Patent Office on December 30, 2000 for Letters Patent, including a true translation of the Official Certificate of the Application.

Signed on 4th day of August, 2006

LEE, JHEE-YOUNG

Full name of translator

Signature of translator

LINE BLDG. 2-3F 823-30, YEOKSAM-DONG,
KANGNAMKU, SEOUL, 135-080

Post Office Address

【Abstract of the Disclosure】

【Abstract】

1. Technical Field of the Invention As Claimed

The present invention relates to a node structure of a multiway search tree, a search method using the node structure, and a computer-readable recording medium for recording a program that implements the method.

2. Technical Object of the Invention

The present invention provides a node structure of a multiway search tree that can reduce the capacity of a main memory and accelerate a search speed by making a key, a key pointer and a node pointer coincident with the size of a cache line through the use of only one pointer written on a node, regardless of the number of keys used in the node, a search method using the node structure, and a computer-readable recording medium for recording a program that implements the search method.

3. Technical solution of the Invention

The present invention provides a node structure of a multiway search tree used to find out an outgoing interface in a system including a central processing unit for executing a routing protocol, a forwarding means for forwarding packets, and a switching means for transmitting data to a corresponding link based on information provided from the forwarding means, the node structure which includes: a plurality of keys K; one key pointer Kp; and one node pointer Po, wherein the key pointer Kp includes information on the number of the keys K

and information on the pointer for informing a down node address.

【Applicability of the Invention】

The present invention is used for address search of a router and a switch system.

【Representative Drawing】

FIG. 7

【Index Words】

Multiway search, node structure, key pointer, node pointer

【Title of the Invention】

Node Structure Of Multiway Search Tree, And Search Method Using It

【Brief Description of the drawings】

FIG. 1 shows a typical giga-bit router.

FIG. 2 illustrates a structure of a typical Binary tree.

FIG. 3 is a structure of a Patricia tree performing a path compression for the Binary tree in FIG. 2.

FIG. 4 provides a level compression result for the typical Binary tree.

FIG. 5 depicts a tree structure depending on a typical cache line.

FIG. 6 represents a structure of a typical B-tree.

FIG. 7 describes a node structure for a search using a multiway search tree in accordance with the present invention.

FIG. 8 shows a structure of the node pointer in FIG. 7.

FIGS. 9A and 9B present a search tree adapted to the search using the multiway search tree in accordance with the present invention.

FIG. 10 exemplifies a flow chart of performing the search using the multiway search tree in accordance with the present invention.

【Detailed Description of the Invention】

【Description of the Invention】

【Field of the Invention and Related Art】

The present invention relates to a node structure of a multiway search tree and a search method using the same; and, more particularly, to a method for generating a node structure of a multiway search tree and a search method for accelerating its search speed by reducing a depth of the multiway search tree and a computer readable recording medium in which a program implementing the search method is recorded.

Referring to FIG. 1, there is illustrated a structure of a typical giga-bit routing system, which includes an interface, a forwarding engine (FE) 14, and a switching fabric.

The interface switches a packet between a data type to be used inside the system and a data type to be used in a link. The forwarding engine (FE) 14 forwarding the packet classifies and assembled a received packet. That is, the forwarding engine 14 finds out a destination of the packet and determines through which link the packet should be outputted. The switching fabric 12 practically provides data onto a link based on information determined at the forwarding engine 14.

As address search methods, there are an exact match and a longest prefix match (LPM) according to how an address is used. The exact match is used in case all bits of an address consist with each other like an Ethernet medium access control (MAC) address. On the other hand, the longest prefix match finds out a node that has the longest bits, starting from the first bit, coincident with a desired key, and, thus, is used in Internet

Protocol version 4 (Ipv4).

Since, however, the longest prefix match finds out the node which is substantially coincident with a comparison object, the searched node is deep so that it takes a long time for searching the node. In order to overcome the above drawback, a Patricia tree has been widely used.

Referring to FIG. 2, there is shown a general binary tree, which is constituted by allocating a smaller value to a left side of a tree and a larger value to a right side of the tree by using a root node as a reference. Meanwhile, FIG. 3 provides a Patricia tree derived from the Binary tree in FIG. 2.

The binary tree is constructed by following a tree according to a bit value of a key, i.e., 0 or 1, starting from the root node until reaching a leaf node. A node met at the end of this process is a node having information consistent with the desired key.

However, in case of the Binary tree, since nodes should be accessed as many as a length of a key, there is a problem in which a path length becomes longer. Therefore, there have been introduced several methods to reduce the path length of the Binary tree.

FIG. 3 is a structure of a Patricia tree performing a path compression for the Binary tree in FIG. 2, FIG. 4 is a diagram of a level compression result for the typical Binary tree.

Unlike the node compression, as illustrated in FIG. 5,

Lampson applies a multiway search tree to an Internet protocol requiring the longest prefix match.

FIG. 5 shows a multiway search tree having 32 bytes of cache lines. Each node consists of keys K1, K2,..., pointers P0, P1, P2,..., and key pointers PK1, PK2,..., wherein each node is composed of 32 bytes.

Lampson uses the multiway search tree so as to maximally utilize a hierarchical memory structure typically employed in modern processors. That is, in order to overcome a difference between a deepening memory speed and a processor speed, a cache having bigger capacity is getting embodied in the processor. According to the operation of the cache, when an arbitrary memory address is accessed, all of cache lines including the memory address are copied onto the cache of the processor at a time and data included in the cache are processed in the processor speed. Therefore, when each node of the multiway search tree is made in the size of one cache line, the number of branches can be substantially increased without using an additional main memory.

Since the multiway search tree is designed by considering a hierarchical memory structure of a disc and a main memory, a disc block provided at a time from the disc to the main memory is used by regarding a low-speed memory and a relatively high-speed main memory. Accordingly, the number of branches should be increased depending on the speed of the main memory without accessing an additional low-speed disc and representative multiway search trees are B-tree and its varied trees.

FIG. 6 shows a tree structure of typical B-tree. Referring to FIG. 6, the B-tree is derived by modifying the Binary tree and usually used in a search algorithm. Namely, like the Binary tree, the B-tree classifies values onto right and left according to their sizes after comparing the values with the root node, whereas, in a next node, the B-tree compares several values at a time according to their sizes so that it can accelerate the search speed and reduce the depth of the tree.

However, in a general case of using the B-tree, the B-tree is also used to construct an index file stored in the disc so as to access a database stored in the disc and the key for accessing the database is configured with a sequence of words so that much more memories are occupied compared with a length of a pointer indicating a next node which will be connected according to a comparison result at a current node.

【Summary of the Invention】

It is, therefore, an object of the present invention to provide a node structure of a multiway search tree that can reduce the capacity of a main memory and accelerate a search speed by making a key, a key pointer and a node pointer coincident with the size of a cache line through the use of only one pointer written on a node, regardless of the number of keys used in the node, a search method using the node structure, and a computer-readable recording medium for recording a program that implements the search method.

In accordance with an aspect of the present invention, there is provided a node structure of a multiway search tree used to find out an outgoing interface in a system including a central processing unit for executing a routing protocol, a forwarding means for forwarding packets, and a switching means for transmitting data to a corresponding link based on information provided from the forwarding means, the node structure includes: a plurality of keys K; one key pointer Kp; and one node pointer Po, wherein the key pointer Kp includes information on the number of the keys K and information on the pointer for informing a down node address.

In accordance with another aspect of the present invention, there is provided a method for searching a multiway search tree to find out an outgoing interface based on address lookup, the method which includes the steps of: a) comparing an inputted IP address with a key value; b) if the inputted IP address is consistent with the key value, searching an outgoing interface by using a key pointer included in the node; c) if the inputted IP address is not consistent with the key value, determining a type of the node by searching a node pointer, and if the node is a leaf node, searching the outgoing interface by acquiring the key pointer after monitoring where the consistency occurs, otherwise, if the node is not the leaf node, moving to a next node with reference to the node pointer, and then repeating the steps of a) to c).

In accordance with another aspect of the present

invention, there is provided a computer-readable recording medium for recording a program that implements a method for searching a multiway search tree to find out an outgoing interface based on address lookup, including the steps of: a) comparing an inputted IP address with a key value; b) if the inputted IP address is consistent with the key value, searching an outgoing interface by using a key pointer included in the node; c) if the inputted IP address is not consistent with the key value, determining a type of the node by searching a node pointer, and if the node is a leaf node, searching the outgoing interface by acquiring the key pointer after monitoring where the consistency occurs, otherwise, if the node is not the leaf node, moving to a next node with reference to the node pointer, and then repeating the steps of a) to c).

Other objects and aspects of the invention will become apparent from the following description of the embodiments with reference to the accompanying drawings, which is set forth hereinafter.

FIG. 7 describes a node structure for a search using a multiway search tree in accordance with the present invention.

Referring to FIG. 7, there is provided a node configuration constituting an 8-way search tree by applying a node structure to a B-tree. That is, one node is composed of 32 bytes and includes 7 keys K1 to K7, one node pointer Po and a key pointer Kp so that the node is coincident with 32 bytes of a cache line. Herein, each of keys is composed of 32 bits

and each pointer is made of 16 bits.

The key pointer Kp is a pointer representing to the first key of the node. In case of a child node, the key pointer represents information for a corresponding port number when a value of the child node is coincident with a key value. In case of no child node, the key pointer depicts region information capable of implementing the longest prefix match.

FIG. 8 shows a structure of the node pointer in FIG. 7. Referring to FIG. 8, the node pointer Po includes both of the number of keys and information for the node pointer. A three-bit most significant bit (MSB) of the node pointer represents the number of keys included in the node and a 13-bit least significant bit (LSB) acts as a pointer reporting a location of a lower level of the tree. Herein, if all values of the node pointer are "1," the node presents a leaf node. On the other hand, if all values of the node pointer are not "1," the node depicts either a root node or a child node, and is used as a pointer for searching a next node.

Referring to FIGS. 9A and 9B, there are illustrated search trees adapted to a search using the multiway search tree in accordance with the present invention.

FIG. 9A shows an example of an 8-way search tree. Assuming that the 8-way search tree is composed of 3 prefixes such as 1*, 1001* and 10100*, 6 keys included in a corresponding range are generated by utilizing the 3 numbers of prefixes as a reference. Herein, the 6 keys are 1* to 100000, 111111, 1001* to 100100, 100111, and 10100* to 101000,

101111.

In FIG. 9B, there is provided a structure of a tree generated by using 6 keys as references. Since the structure carries out the longest prefix match, a port number is assigned to each region. Accordingly, if there is found a consistent key as a result of searching from the root node to the leaf node, information for the port can be obtained. On the other hand, if there is no consistent key in the leaf node, the information for the port can be obtained from the corresponding range.

For instance, it a packet whose IP address is 110000 is arrived, the packet comes under a 6thkey pointer between 101111 and 111111, the address of the packet is located at P1 port. On the other hand, if the inputted IP address consists with the key, the key pointer is directly obtained from Kp and location information N.

FIG. 10 describes a flow chart of representing a search using the multiway search tree in accordance with a preferred embodiment of the present invention.

Referring to FIG. 10, in steps 101 and 102, after reading out an 8-way node through a search operation, the IP address of the node is compared with 7 numbers of key values.

As a comparison result of step 102, if the IP address is consistent with the key values, in step 103, a destination corresponding to the IP address is found by using the key pointer obtained from the comparison result. On the other hand, if the IP address is not consistent with the key values, in

step104, a value of the node pointer is read out. In step105, it is determined whether the node pointer corresponds to a child node or a leaf node.

As a determination result of step 105, if it is decided that the node pointer corresponds to the leaf node, in step106, a range of the keys is acquired by referring to the above comparison result. Then, in step107, the destination is found by calculating a key pointer.

Meanwhile, as a result of step 105, if it is determined that the node pointer corresponds to the child node, the region to which the keys correspond is searched in step 108.

In order to move to a next node, a next node pointer is computed by using its corresponding pointer and then the search process moves to the next node in step 109. After then, the comparison process of step 102 is performed and this search process is repeated until the consistent key value is found at the leaf node or the leaf node is reached.

The present invention described above can be implemented to a program capable of being stored in a computer readable recording medium such as a compact disk read only memory (CD-ROM), a random access memory (RAM), a read only memory (ROM), a floppy disc, a hard disc, an optical magnetic disc and the like.

While the present invention has been described with respect to the particular embodiments, it will be apparent to those skilled in the art that various changes and modifications may be made without departing from the spirit

and scope of the invention as defined in the following claims.

【Effects of the Invention】

As illustrated above, the present invention can increase the number of keys capable of being recorded on a cache line by using one pointer at a node of the multiway search tree so that the number of branches in a network address search is also increased and thus the tree depth is reduced. As a result, the present invention can accelerate the search speed and the speed of the forwarding engine.

Further, the present invention can accomplish a further speed-up by decreasing required memories and thus increasing a memory rate used in a second cache.

【Claims】

【Claim 1】

A node structure of a multiway search tree used to find out an outgoing interface in a system including a central processing unit for executing a routing protocol, a forwarding means for forwarding packets, and a switching means for transmitting data to a corresponding link based on information provided from the forwarding means, comprising:

a plurality of keys K;

one key pointer Kp; and

one node pointer Po,

wherein the key pointer Kp includes information on the number of the keys K and information on the pointer for informing a down node address.

【Claim 2】

The node structure as recited in claim 1, wherein when the node pointer Po is all composed of '1,' a corresponding node is a leaf node.

【Claim 3】

The node structure as recited in claim 1 or 2, wherein each of the keys are composed of 32 byts and the key pointer Kp is composed of 16 bits.

【Claim 4】

The node structure as recited in claim 1 or 2, wherein the number of keys K is 7 to form an 8-way search tree.

【Claim 5】

The node structure as recited in claim 1 or 2, wherein when a leaf node is formed, a key pointer is allocated to each key region to achieve a largest prefix matching.

【Claim 6】

A method for searching a multiway search tree to find out an outgoing interface based on address lookup, comprising the steps of:

a) comparing an inputted IP address with a key value;

b) if the inputted IP address is consistent with the key value, searching an outgoing interface by using a key pointer included in the node;

c) if the inputted IP address is not consistent with the key value, determining a type of the node by searching a node pointer, and if the node is a leaf node, searching the outgoing interface by acquiring the key pointer after monitoring where the consistency occurs, otherwise, if the node is not the leaf node, moving to a next node with reference to the node pointer, and then repeating the steps of a) to c).

【Claim 7】

A computer-readable recording medium for recording a program that implements a method for searching a multiway search tree to find out an outgoing interface based on address lookup, comprising the steps of:

a) comparing an inputted IP address with a key value;

b) if the inputted IP address is consistent with the key value, searching an outgoing interface by using a key pointer included in the node;

c) if the inputted IP address is not consistent with the key value, determining a type of the node by searching a node pointer, and if the node is a leaf node, searching the outgoing interface by acquiring the key pointer after monitoring where the consistency occurs, otherwise, if the node is not the leaf node, moving to a next node with reference to the node pointer, and then repeating the steps of a) to c).

FIG. 1

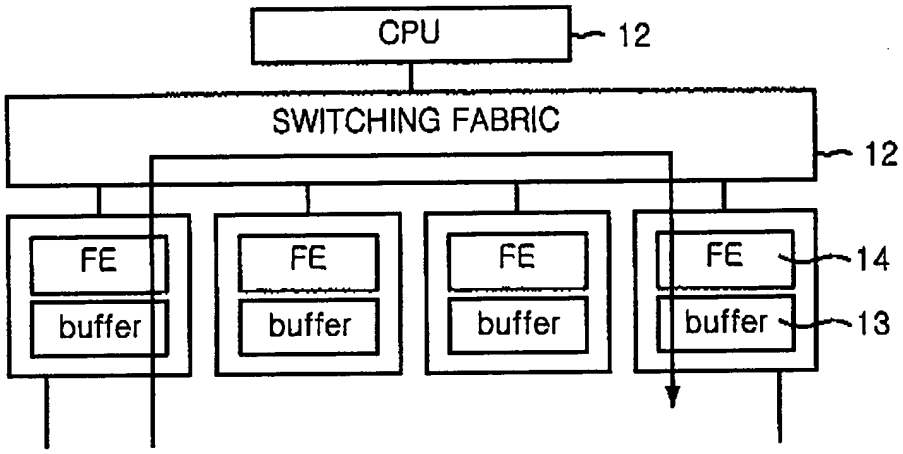

CPU
12
SWITCHING FABRIC
12
FE
buffer
FE
buffer
FE
buffer
FE
buffer
14
13

FIG. 2

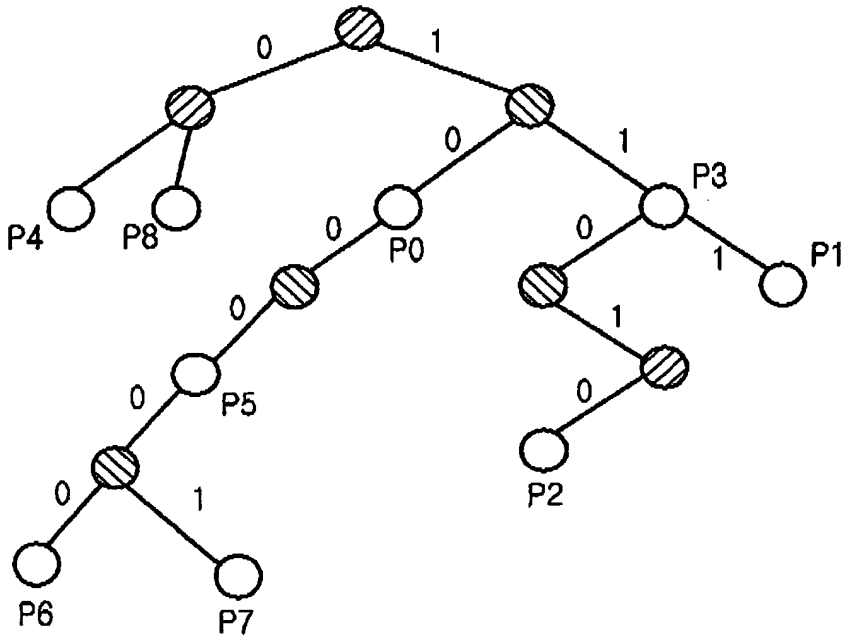


FIG. 3

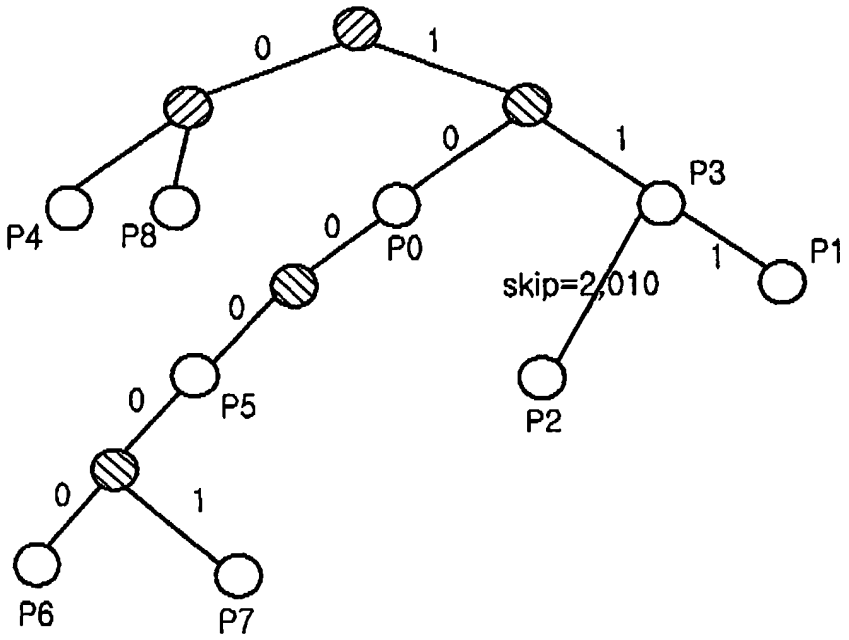

0
1
0
1
P4
P8
0
P0
P3
0
skip=2,010
1
P1
0
P5
P2
0
1
P6
P7

FIG. 4

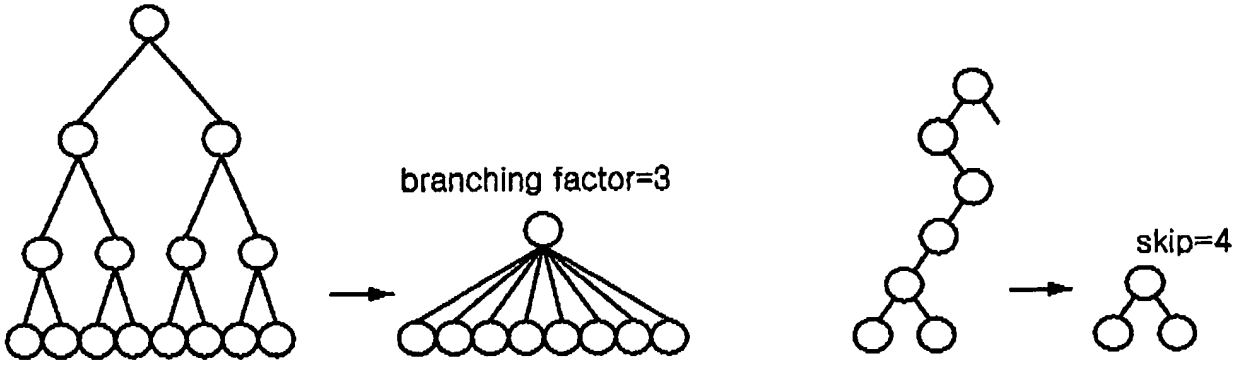

branching factor=3
skip=4

FIG. 5

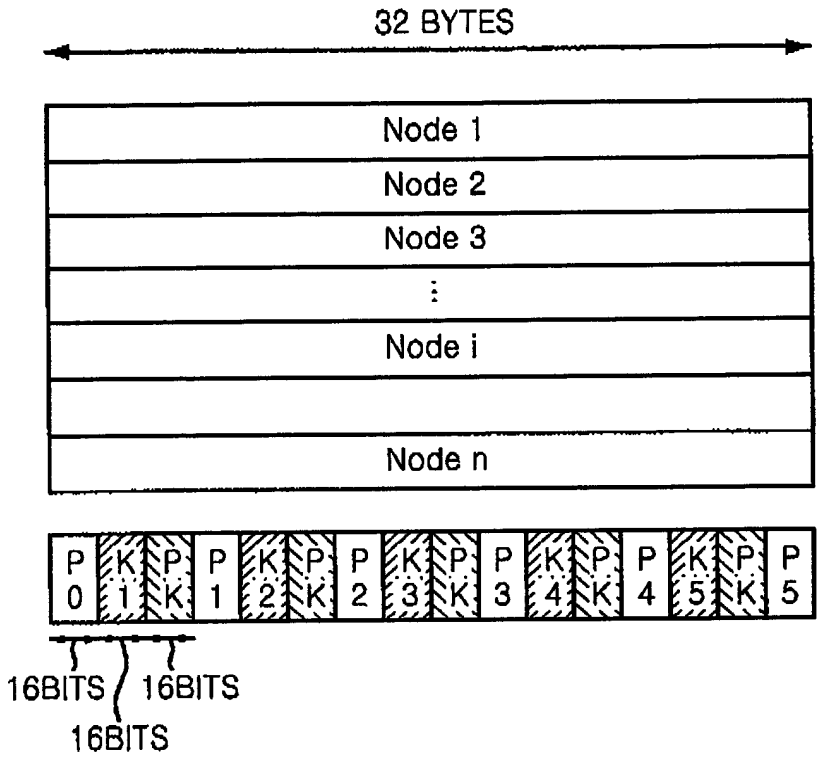

32 BYTES
Node 1
Node 2
Node 3
Node i
Node n
P0
K1
PK
P1
K2
PK
P2
K3
PK
P3
K4
PK
P4
K5
PK
P5
16BITS
16BITS
16BITS

FIG. 6

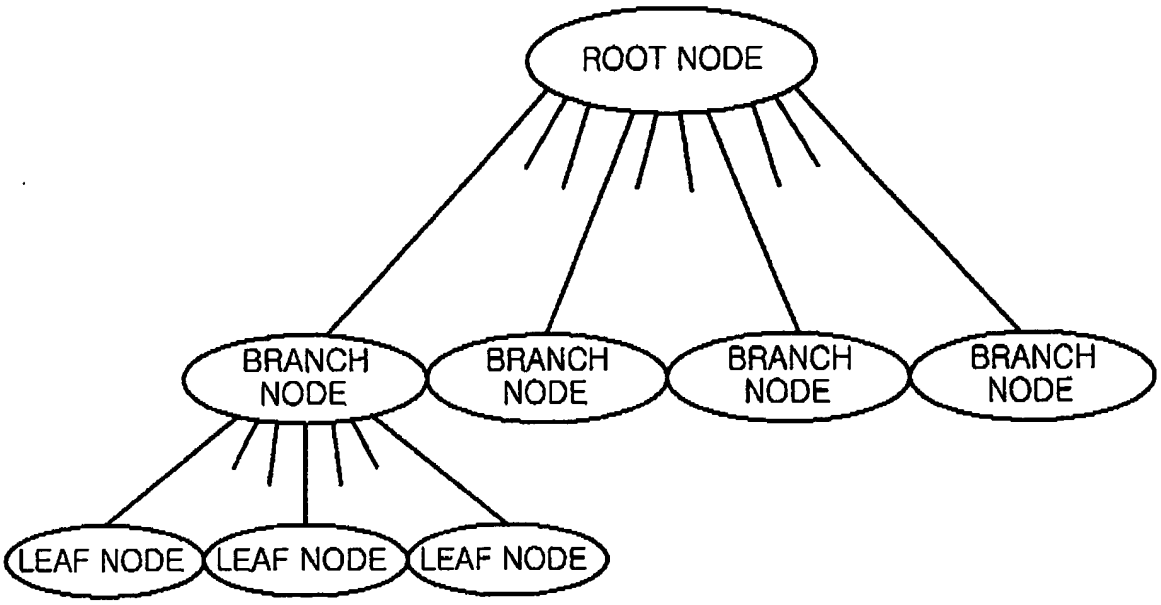

ROOT NODE
BRANCH NODE
BRANCH NODE
BRANCH NODE
BRANCH NODE
LEAF NODE
LEAF NODE
LEAF NODE

FIG. 7

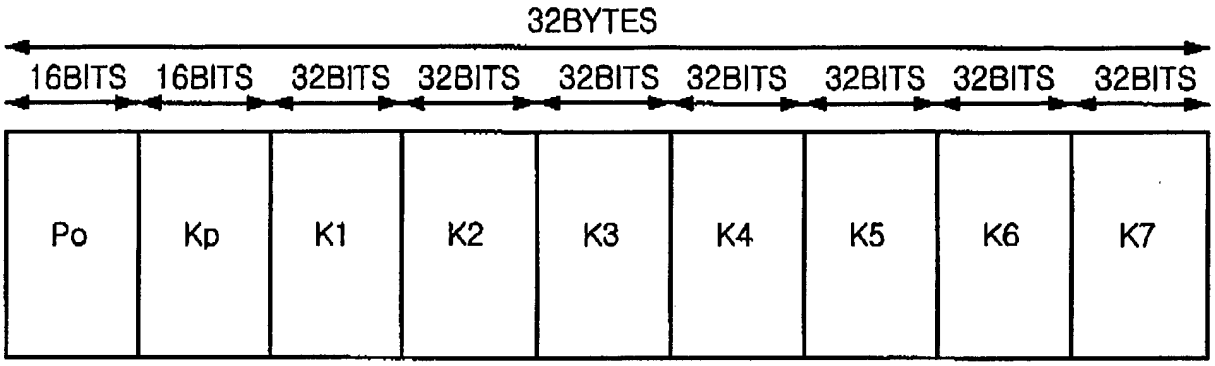

32BYTES
16BITS
16BITS
32BITS
32BITS
32BITS
32BITS
32BITS
32BITS
32BITS
Po
Kp
K1
K2
K3
K4
K5
K6
K7

FIG. 8

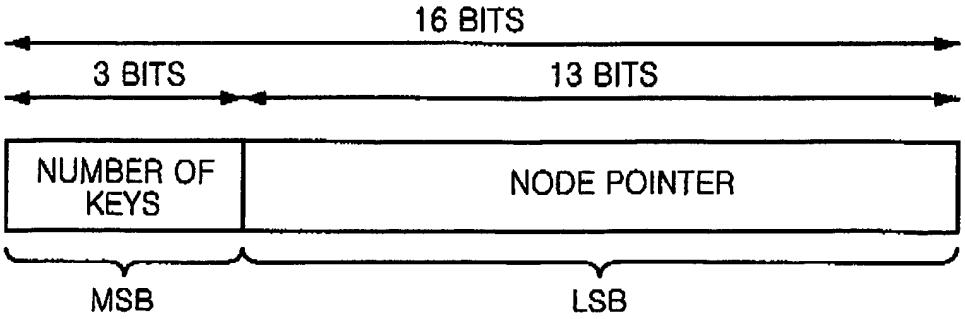

16 BITS
3 BITS
13 BITS
NUMBER OF KEYS
NODE POINTER
MSB
LSB

FIG. 9A

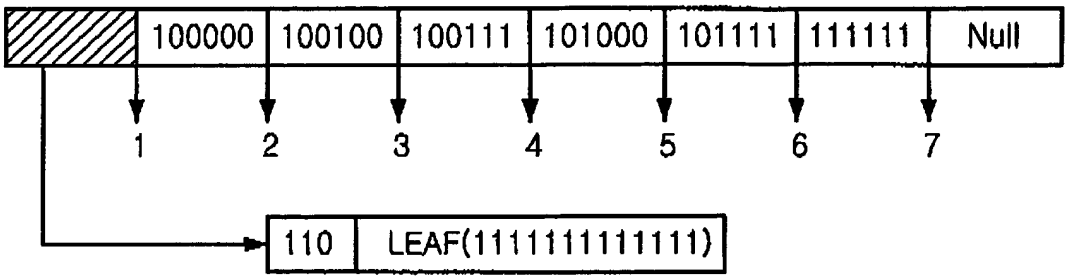

100000
100100
100111
101000
101111
111111
Null
1
2
3
4
5
6
7
110
LEAF(1111111111111)

FIG. 9B

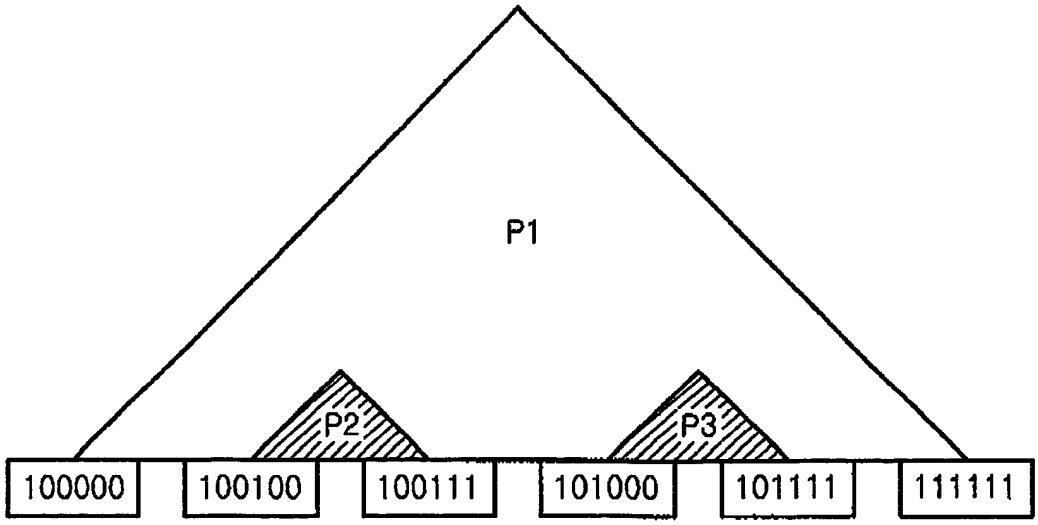
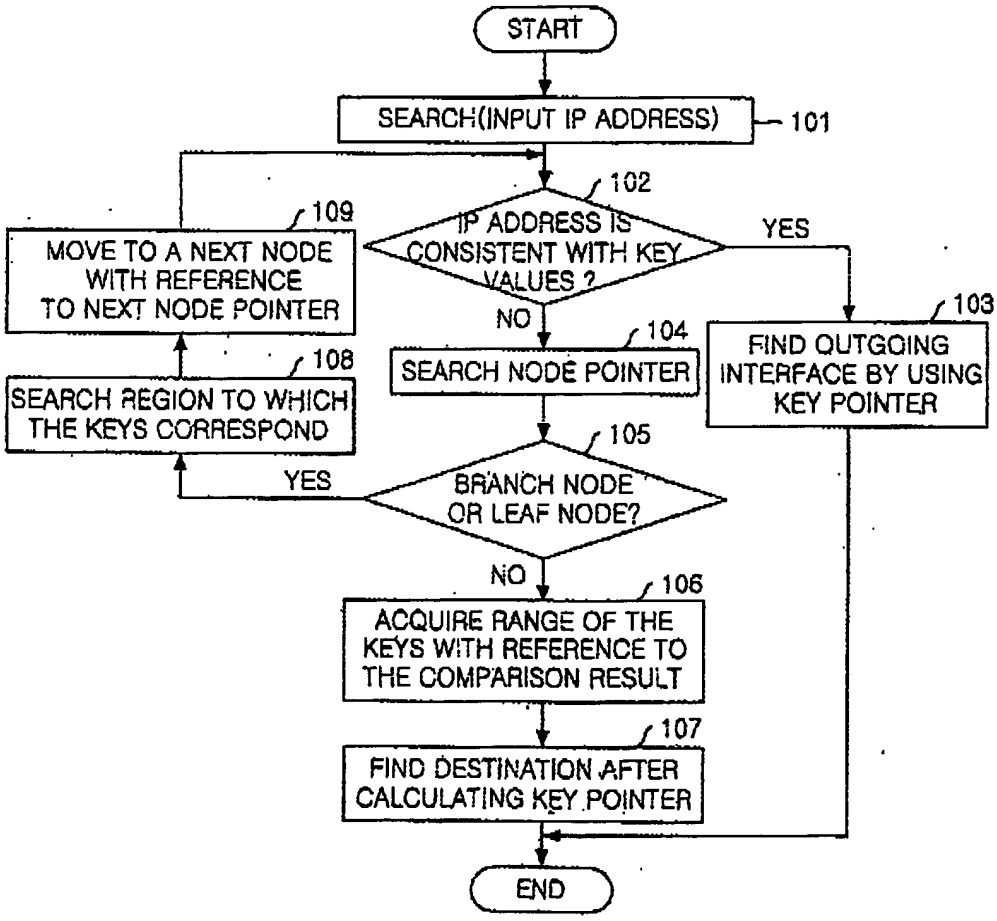

P1
P2
P3
100000
100100
100111
101000
101111
111111

FIG. 10


START
SEARCH(INPUT IP ADDRESS)
101
102
IP ADDRESS IS CONSISTENT WITH KEY VALUES ?
YES
NO
103
FIND OUTGOING INTERFACE BY USING KEY POINTER
104
SEARCH NODE POINTER
105
BRANCH NODE OR LEAF NODE?
YES
NO
106
ACQUIRE RANGE OF THE KEYS WITH REFERENCE TO THE COMPARISON RESULT
107
FIND DESTINATION AFTER CALCULATING KEY POINTER
108
SEARCH REGION TO WHICH THE KEYS CORRESPOND
109
MOVE TO A NEXT NODE WITH REFERENCE TO NEXT NODE POINTER
END